



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/749,938	12/29/2000	Donald Brent Marshall	56130.000043	6808
7590	07/07/2004		EXAMINER	
Hunton & Williams 1900 K Street, N.W. Washington, DC 20006-1109			INGBERG, TODD D	
			ART UNIT	PAPER NUMBER
			2124	7
DATE MAILED: 07/07/2004				

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)
	09/749,938	MARSHALL ET AL. 
Examiner	Art Unit	
Todd Ingberg	2124	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1) Responsive to communication(s) filed on 08 April 2004.

2a) This action is **FINAL**. 2b) This action is non-final.

3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4) Claim(s) 1-48 is/are pending in the application.
4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5) Claim(s) _____ is/are allowed.

6) Claim(s) 1-48 is/are rejected.

7) Claim(s) _____ is/are objected to.

8) Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9) The specification is objected to by the Examiner.

10) The drawing(s) filed on _____ is/are: a) accepted or b) objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
a) All b) Some * c) None of:
1. Certified copies of the priority documents have been received.
2. Certified copies of the priority documents have been received in Application No. _____.
3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) Notice of References Cited (PTO-892)
2) Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____
4) Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____
5) Notice of Informal Patent Application (PTO-152)
6) Other: _____

DETAILED ACTION

Claims 1 – 48 have been examined.

Claims 47 and 48 have been added.

Drawings

1. The drawings submitted on March 6, 2001 were received. The drawing are objected to by the PTO Draftsperson as indicated on PTO-948. Corrections to drawings are required. No response was received.

Specification

2. The amendment to the Specification showing the Cross Related Applications has been entered.

3. The new title has been entered.

Claim Rejections - 35 USC § 102

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

1. Claims 1 – 46 are rejected under 35 U.S.C. 102(b) based upon a public use or sale of the invention. Template Software Corporation's commercial product "SNAP 8.0", released in 1997.

2. Template Software. The Template product line is object oriented and contains the SNAP programming language and the Workflow Template (WFT). The documentation sets for the products contain the following manuals.

SNAP released June 1997

SNAP Language Reference (Not used in this Office Action)

Using the SNAP Language (Not used in this Office Action)

Using the SNAP Communication Component (Referred to as **COMM**)

Using the SNAP Graphic User Interface Component (Not used in this Office Action)

Getting Started with SNAP (Not used in this Office Action)

Using the SNAP Display Editors (Not used in this Office Action)

SNAP Class Library Reference (Not used in this Office Action)

Using the SNAP External Application Software Component (Not used in this Office Action)

Using the SNAP Development Environment (Referred to as **SNAP**)

SNAP Module Library Reference (Not used in this Office Action)

Using the SNAP Permanent Storage Component (Referred to as **PERM**)

Workflow released September 1997

Developing a WFT Workflow System (Not used in this Office Action)

Using the WFT Development Environment (Not used in this section of the Office Action)

WFT Library Reference (Not used in this Office Action)

Since, these products work together they constitute a single reference and can be used as the basis for a rejection based on anticipated by a product offering.

Claim 1

Template anticipates a system for providing an application component where the application component enables a service to be managed as an independent entity comprising (**Template** in the **SNAP** manual, Chapters 5 and 6, cover the building of Shared Information Base (SIB) filter functions and many different schemas with attribute editing functions (**SNAP**, page 5-19, Import and Export maps – the method calls of the object oriented CASE tool which performs the gets and sets of the maps of the SIB functions and the Schema editor functions setting up schemas also see (The façade is an object which performs the SIB applications **COMM**, pages 4-11 to 4-15, page 4-19 to 4-20, page 4-22 to 4-32 also page 8-1 to 8-2 filter functions): a context for containing logic and data associated with a service session; a facade for containing context-independent service logic wherein the facade is not associated with the service session (the facade is the ability to create a object oriented template that is instantiated when SIB connection is made to perform filter functions and further supported by the ability to make many different views with schemas); and an event portal for providing entry and exit interfaces. (**Template** in the **SNAP** manual, Chapter 6, cover the building of Shared Information Base (SIB) for mapping to the many different schemas. The SIB connection has filter functions which are at the application layer and are not protocol dependent **COMM** Chapter 5 the application layer which is protocol independent for mapping attributes and Chapter 8 of **COMM** the filter functions, – Also relevant is the mapping of the schema to the application as taught in chapter 6 – note the different schemas are not tied to a specific entry but reflect the view the application has to the database also **COMM**, pages 4-17 to 4-32).

Claim 2

The system of claim 1 further comprising management logic for defining operations, administration and management behavior. (SNAP, on page 5-3 the terms are defined – page 5-6 the basic built in functions as well as the filter functions and attribute editing of claim 1).

Claim 3

The system of claim 1 further comprising management logic for defining appearance of the application component (SNAP, page 5-8, SIB Mapping see page 5-12 to 5-13 of COMM for autogets and autosets and COMM, pages 4-17 to 4-32 and chapter 8).

Claim 4

The system of claim 1 further comprising a wiring tool to configure a connection between the event portal of the application component to another event portal of a second application component. (SNAP, page 5-19, Import and Export maps – the method calls of the object oriented CASE tool which performs the gets and sets of the maps of the SIB functions and the Schema editor functions).

Claim 5

The system of: claim 4 wherein the wiring tool connects one or more outgoing events from the event portal to one or more incoming events of an event portal associated with the entity. (SNAP, Chapter 6 Introductions, schema editors ability to communicate to databases via SIB and to COMM Chapter 5 the application layer which is protocol independent for mapping attributes and Chapter 8 of COMM the filter functions).

Claim 6

The system of: claim 4 wherein the wiring tool provides the ability to create service variants by modifying connections between application components (Chapter 8 of **COMM** the filter functions and **COMM**, pages 4-17 to 4-32).

Claim 7

The system of claim 4 wherein the connection does not require hardcoding thereby enhancing flexibility in changing connections. (SNAP, page 5-8, SIB editor parameter environment).

Claim 8

The system of claim 4 wherein wiring definitions are uploaded to a service execution engine wherein the service execution engine creates the connection. (SNAP, SIB executing by definition working with schema).

Claim 9

The system of claim 1 wherein the application component is network independent. SNAP, SIB, different SIB connectors are built in for different connectors. Application connection through SNAP to SIB page 5-2 Inter process Communications and SNAP chapter 6 many different databases).

Claim 10

The system of claim 1 wherein the application component encapsulates protocol specific interactions and presents a homogenous interface to other components. (SNAP and **COMM**, SIB for handling filter functions and attribute editing and **COMM**, pages 4-17 to 4-32 as per above).

Claim 11

The system of claim 1 wherein the application component is network independent and protocol independent. (SNAP, is at the application layer as are the SIB functions which perform filter functions and attribute editing as per above).

Claim 12

The system of claim 4 wherein the connection is postponed until after the application component is created. Interpreted as execution of SIB in object oriented environment the object control for SIB must be instantiated as per claim 8.

Claim 13

The system of claim 1 wherein the service comprises more than one application component where application components are developed by separate developers thereby enabling parallel development. (SNAP, pages 2-49 to 2-51 version control).

Claim 14

The system of claim 4 wherein runtime context event subscription is established dynamically based on static event subscription definition. (SNAP, page 6-24, filter function **and COMM**, pages 4-17 to 4-32).

Claim 15

The system of claim 4 wherein contexts of different application components pertaining to a service session are maintained in a context envelope. (SNAP and COMM, SIB as taught by the applications that maintain the filter functions and attribute editing functions per above – these are referred to as methods).

Claim 16

The system of claim 15 wherein the contexts are added dynamically to the context envelope as the contexts are invoked by service logic. (SNAP, page 6-24, filter function).

Claim 17

The system of claim 1 wherein one or more service variants are selected by the facade for each service session. As per claim 17.

Claim 18

The system of claim 17 wherein the application component contains a single template which executes by default. (SNAP, defining only 1 SIB connector – the portion of the SIB being claimed is at the application layer).

Claim 19

The system of claim 1 wherein the application component incorporates one or more of data storage schemas, variables, constants and configuration items. (SNAP, chapter 6, page 6-2 to 6-5 Schema Editor and COMM, pages 4-17 to 4-32, filter functions and autogets and autosets as per above).

Claim 20

The system of claim 19 wherein the one or more of data storage schemas, variables, constants and configuration items are exported from the application component. (SNAP, as per claim 19, pages 6-7 to 6-13 and COMM filter functions and autogets and autosets as per above).

Claim 21

The system of claim 19 wherein the one or more of data storage schemas, variables, constants and configuration items are imported by the application component. As per claims 19 and 20.

Claim 22

The system of claim 20 wherein a wiring tool connects an exported item from the application component with an imported item in another application component. (SNAP, pages 6-19 and 6-20, filter callback – the execution of these methods in the object oriented environment causes messages to be sent).

Claim 23

The system of claim 1 wherein one or more protocol specific interactions are encapsulated to present a homogenous interface to other one or more application components. (SNAP and COM, as per claim 10).

Claim 24

Template anticipates a method for providing an application component where the application component enables a service to be managed as an independent entity (an object) comprising the steps of: maintaining logic and data associated with the service in a context; maintaining context -independent service logic in a facade wherein the facade is not associated with the service; and providing entry and exit interfaces. See the rejection for claim 1.

Claim 25

The method of claim 24 further comprising the step of enabling management logic to define operations, administration and management behavior. See the rejection for claim 2.

Claim 26

The method of claim 24 further comprising the step of enabling management logic to defining appearance of the application component. See the rejection for claim 3.

Claim 27

The method of claim 24 further comprising the step of providing a wiring tool to configure a connection between the event portal of the application component to another event portal of a second application component. See the rejection for claim 4.

Claim 28

The method of claim 27 wherein the wiring tool connects one or more outgoing events from the event portal to one or more incoming events of an event portal associated with the entity. See the rejection for claim 5.

Claim 29

The method of claim 27 wherein the wiring tool provides the ability to create service variants by modifying connections between application components. See the rejection for claim 6.

Claim 30

The method of claim 27 wherein the connection does not require hardcoding thereby enhancing flexibility in changing connections. See the rejection for claim 7.

Claim 31

The method of claim 27 wherein wiring definitions are uploaded to a service execution engine wherein the service execution engine creates the connection. See the rejection for claim 8.

Claim 32

The method of claim 24 wherein the application component is network independent. See the rejection for claim 9.

Claim 33

The method of claim 24 wherein the application component encapsulates protocol specific interactions and presents a homogenous interface to other components. See the rejection for claim 10.

Claim 34

The method of claim 24 wherein the application component is network independent and protocol independent. See the rejection for claim 11.

Claim 35

The method of claim 27 wherein the connection is postponed until after the application component is created. See the rejection for claim 12.

Claim 36

The method of claim 24 wherein the service comprises more than one application component where application components are developed by separate developers thereby enabling parallel development. See the rejection for claim 13.

Claim 37

The method of claim 27 wherein runtime context event subscription is established dynamically based on static event subscription definition. See the rejection for claim 14.

Claim 38

The method of claim 27 wherein contexts of different application components pertaining to a service session are maintained in a context envelope. See the rejection for claim 15.

Claim 39

The method of claim 38 wherein the contexts are added dynamically to the context envelope as the contexts are invoked by service logic. See the rejection for claim 16.

Claim 40

The method of claim 24 wherein one or more service variants are selected by the facade for each service session. See the rejection for claim 17.

Claim 41

The method of claim 40 wherein the application component contains a single template which executes by default. See the rejection for claim 18.

Claim 42

The method of claim 24 wherein the application component incorporates one or more of data storage schemas, variables, constants and configuration items. See the rejection for claim 19.

Claim 43

The method of claim 42 wherein the one or more of data storage schemas, variables, constants and configuration items are exported from the application component. See the rejection for claim 20.

Claim 44

The method of claim 42 wherein the one or more of data storage schemas, variables, constants and configuration items are imported by the application component. See the rejection for claim 21.

Claim 45

The method of claim 42 wherein a wiring tool connects an exported item from the application component with an imported item in another application component. See the rejection for claim 22.

Claim 46

The method of claim 24 wherein one or more protocol specific interactions are encapsulated to present a homogenous interface to other one or more application components. See the rejection for claim 23.

Claim 47

Template anticipates a system for providing an application component where the application component enables a service to be managed as an independent entity (SIB provides an application layer for filter functions **COMM**, pages 4-17 to 4-32 and for autoget and auto set functions for Schemas also see the rejection for claim 1), comprising: a context for containing logic and state data associated with a transaction (**COMM**, pages 4-11 to 4-15, page 4-19 to 4-20, page 4-22 to 4-32 also page 8-1 to 8-2 filter functions), , the context having a plurality of variants wherein each context variant is associated with a specific transaction (as per above - the ability to get or set or use a filter function to change the value depending on the method called) ; a facade for containing context-independent service logic wherein the facade is not associated with the transaction wherein the facade instantiates a plurality of context variants depending on configuration data (The façade is an object which performs the SIB applications **COMM**, pages 4-11 to 4-15, page 4-19 to 4-20, page 4-22 to 4-32 also page 8-1 to 8-2 filter functions) ; and an event portal for providing entry and exit interfaces for the application component wherein the event portal sends and receives at least one event for the transaction wherein the at least one event comprises an object used to communicate details of an occurrence (in object oriented programming the entry and exit are part of messaging – an inherent feature of object technology) ; wherein the facade processes the event for the transaction and invokes a specific context variant of the plurality of context variants and adds the specific context variant to a context envelope for establishing a transaction specific communication path (SNAP and **COMM**, SIB as taught by the applications that maintain the filter functions and attribute editing functions per above – these are referred to as methods – the specific path is the message in object oriented technology from the sender to the receiver as per the rejections), and wherein the application component is protocol independent and network independent (The SIB functions used are at the application layer not the protocol layer this is independent of the network and the protocol).

Claim 48

Template anticipates a method for providing an application component where the application component enables a service to be managed as an independent entity comprising the steps of: maintaining logic and state data associated with a transaction in a context, the context having a plurality of variants wherein each context variant is associated with a specific transaction; maintaining context-independent service logic in a facade wherein the facade is not associated with the transaction (a method to be used regardless of who calls the method) wherein the facade instantiates a plurality of context variants depending on configuration data; and providing entry and exit interfaces for the application component wherein the event portal sends and receives at least one event for the transaction wherein the at least one event comprises an object used to communicate details of an occurrence; wherein the facade processes the event for the transaction and invokes a specific context variant of the plurality of context variants and adds the specific context variant to a context envelope for establishing a transaction specific

communication path, and wherein the application component is protocol independent and network independent. As per the rejection of claim 47.

Response to Arguments

Applicant's Argument

"Claim Rejections - 35 U.S.C. § 102(b)

Claims 1-46 are currently rejected under 35 U.S.C. § 102(b) based on a public use or sale by Template Software Corporation's commercial product SNAP 8.0, released in 1997. More specifically, the Office Action relies upon the "Using the SNAP Development Environment" manual (hereinafter "SNAP reference").

Chapter 5 of the SNAP reference discusses the Shared Information Base (SIB) facilities that supports interprocess communication. SIB facilities are used to share classes among the processes in an application. The SIB Connection Editor is used to define communications connections between processes. As stated on 5-4 of the SNAP reference, a SIB connection is the mechanism through which two processes communicate using the TCP/IP communication protocol. The communicating processes connect to share data in the form of classes.

For a proper rejection under 102(b), each and every limitation of the claims must be shown in a single reference. The SNAP reference fails to show each and every limitation as claimed by Applicants. Therefore, the rejection is improper and should be withdrawn.

The SNAP reference fails to show an application component comprising a combination of context, facade and event portal wherein a service is managed as an independent entity. For example, the SNAP reference fails to disclose a facade for containing context-independent service logic wherein the facade is not associated with the service session. This limitation is completely missing from the SNAP reference. Rather, the SNAP reference appears to be concerned with a SIB connection through which two processes communicate using the TCP/IP communication protocol. The Office Action states that the SNAP reference discloses an ability to create an object oriented template but fails to show how the SNAP reference discloses a facade for containing context-independent service logic wherein the facade is not associated with the service session. As the SNAP reference fails to disclose at least this limitation, the rejection is improper and should be withdrawn."

Examiner's Response

Applicant must take the reference as a whole. Applicant has taken the portion of the reference for Chapter 5 and steered the arguments in one direction. Applicant's claims are directed toward the application layer but Applicant is focused on a lower layer of the reference. The Examiner has added more art to the rejection. The application layer is independent of protocol and the focus of the rejection. Visibly absent is the portion for Chapter 6 which covers the Using the Database Mapping Editor. In the broadest reasonable interpretation in view of the Specification the rejection is reasonable. Applicant's Specification page 1 in the Field Of Invention is as follows:

"Field of Invention

The present invention relates generally to software components, and more specifically to flexible service application components that enable a service to be developed, provisioned,

managed and maintained as a separate entity with a well defined external interface, specified by events that are imported and exported and./or the specific points at which interaction occurs with other components.”

The prior art rejection is more than merely a SIB connection. The SIB connection also has the ability to build filter functions and Schema mapping (many different Data Bases – local or distributed) which interact with applications. The functionality of Template is that of an Object Oriented Computer Aided Software Engineering (OO-CASE) tool and middleware product as well as a Workflow system. The combination of these functions meet the current claim limitations. The failure to acknowledge the Data Base Mapping chapter is viewed as a piecemeal attack on the reference. In response to applicant's arguments against the references individually, one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.*, 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986).

The limitations the Applicant says are not taught are “ For example, the SNAP reference fails to disclose a facade for containing context-independent service logic wherein the facade is not associated with the service session. “ The Filter functions of the SIB meets this limitation also well as the Schema mapping which performs editing of attributes such as on page 6-24. Applicant states “This limitation is completely missing from the SNAP reference. Rather, the SNAP reference appears to be concerned with a SIB connection through which two processes communicate using the TCP/IP communication protocol.” This is a narrow view of the reference and is considered piecemeal. Applicant states “The Office Action states that the SNAP reference discloses an ability to create an object oriented template but fails to show how the SNAP reference discloses a facade for containing context-independent service logic wherein the facade

Art Unit: 2124

is not associated with the service session.” The Templates are tools for creating objects which perform functions. The specific functions such as filter functions and attribute editing and monitoring are considered facades. These middleware functions perform a wrapper like function which are session independent. Applicant steers toward a protocol level in the arguments for functions that are protocol independent. The arguments are not deemed persuasive. But the Examiner feels the arguments show the rejection would be stronger if more of the reference is provided showing more detail of the application layer of the SIB facility. The application layer being protocol independent.

Applicant's Argument

“ In addition, the SNAP reference fails to further disclose the context for containing logic and data associated with a service session and an event portal for providing entry and exit interfaces. These limitations are recited in independent claims 1 and 24. “

Examiner's Response

As above the narrow focus on SIB connection is only a portion of the rejection. Applicant should revisit the filter functions and Schema attribute editing and monitoring functions of the reference. Examiner has more clearly rejected the claims with filter functions and attribute editing functions.

Applicant's Argument

“Dependent claims 2-23 and 25-46 further recite additional features that are not disclosed in the SNAP reference. For instance, the SNAP reference fails to disclose the wiring tool, as recited by Applicants. The SNAP reference also fails to show the network independence and protocol independence feature, claimed by Applicants. Rather, the SNAP reference is dependent on the TCP/IP communication protocol. The specific limitations directed to the context are also missing from the SNAP reference.”

Examiner's Response

The limitation wiring tool was interpreted to be the ability to write object oriented programs. The term *wiring tool* often refers to the messaging capabilities of object technology. In terms of the argument for network independence and protocol independence the Template systems operates both standalone or peer-to-peer which meets one interpretation but the critical point is the application functions such as filter functions and attribute editing are at an application level. The Applicant argues the context is missing the filter function must have context just as a schema attribute editing function must have contexts. These are not missing.

Applicant's Argument

“New claims 47 and 48 have been added to further clarify the novel features of the invention. Claim 47 recites “a context for containing logic and state data associated with a

transaction, the context having a plurality of variants wherein each context variant is associated with a specific transaction;" "a facade for containing context-independent service logic wherein the facade is not associated with the transaction wherein the facade instantiates a plurality of context variants depending on configuration data;" and "an event portal for providing entry and exit interfaces for the application component wherein the event portal sends and receives at least one event for the transaction wherein the at least one event comprises an object used to communicate details of an occurrence;" "wherein the facade processes the event for the transaction and invokes a specific context variant of the plurality of context variants and adds the specific context variant to a context envelope for establishing a transaction specific communication path," and "wherein the application component is protocol independent and network independent." Claim 48 includes similar corresponding limitations."

Examiner's Response

New claims have additional grounds for rejection.

Conclusion

Applicant's arguments more clearly limit the invention. Although the original reference teaches the invention the reference is large. The Examiner does not believe making this action FINAL would be proper, instead the Examiner has conformed to the more narrow interpretation provided in the arguments and has realigned the rejection to different aspects of the reference. This is despite the fact the Applicant did not argue the entire rejection. The arguments support the aspects of filter functions and attribute function when connected to a database. Both are independent of protocol and are at the application layer.

Correspondence Information

4. Any inquiry concerning this communication or earlier communications from the examiner should be directed to **Todd Ingberg** whose telephone number is (703) 305-9775. The examiner can normally be reached during the following hours:

Monday	Tuesday	Wednesday	Thursday	Friday
6:15 – 1:30	6:15- 3:45	6:15 – 4:45	6:15-3:45	6:15-130

This schedule began December 1, 2003 and is subject to change.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, **Kakali Chaki** can be reached on (703) 305-9662. Please, note that as of August 4, 2003 the **FAX number** changed for the organization where this application or proceeding is assigned is **(703) 872-9306**.

Also, be advised the United States Patent Office **new address** is

Post Office Box 1450

Alexandria, Virginia 22313-1450

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703) 305-9700.



Todd Ingberg
Primary Examiner
Art Unit 2124
June 28, 2004